Research Heaven,
West Virginia

See More, Learn More, Tell More

Tim Menzies
West Virginia University
tim@timmenzies.net

Problem

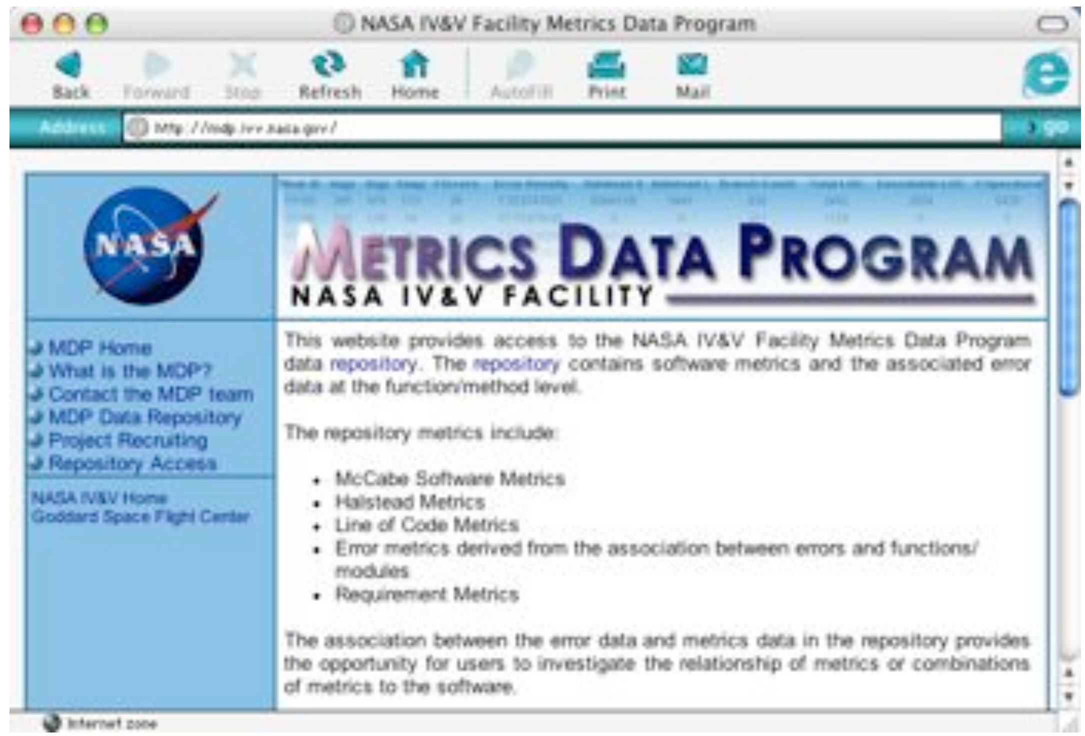
Research Heaven,
West Virginia

- **Mountains of data**
  - Seek “the diamonds in the dust”
- **We have many do-ings**
  - But what are we learn-ing?
- **What general lessons about software quality assurance can we offer NASA?**
- **Problem of external validity**
  - It worked “there” but will it work “here”?

Approach

Research Heaven,
West Virginia

- **while not ((end of time OR end of money))**
  - chase data sets
  - extract cost-benefit patterns from data
  - check the stability of those patterns
  - report stable conclusions
- **Product metrics:**
  - NASA metric's data program
  - Goddard project
  - Flight simulators
- **Process metrics:**
  - cost estimation data from JPL
    - Now spun off into a project with Jairus Hihn
  - SILAP (IV&V effort potential model)



http://mdp.ivv.nasa.gov

Now with 10+ projects ⇐ and many more soon

Importance/ Benefits

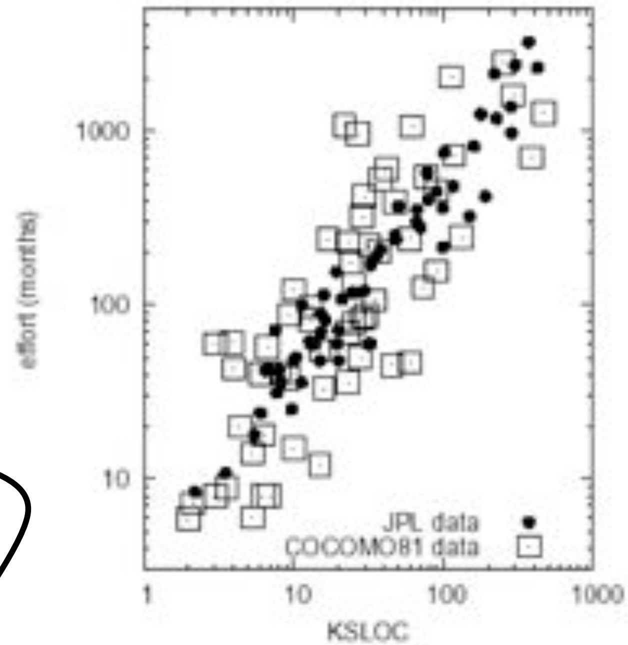
Research Heaven,
West Virginia

- **Generally:**
  - –NASA does a lot of software
  - –What guidance should we offer developers?
  - –How good is that guidance
    - Has that guidance been certified?
    - Do we know how general are those guidelines?

Relevance To NASA

Research Heaven,

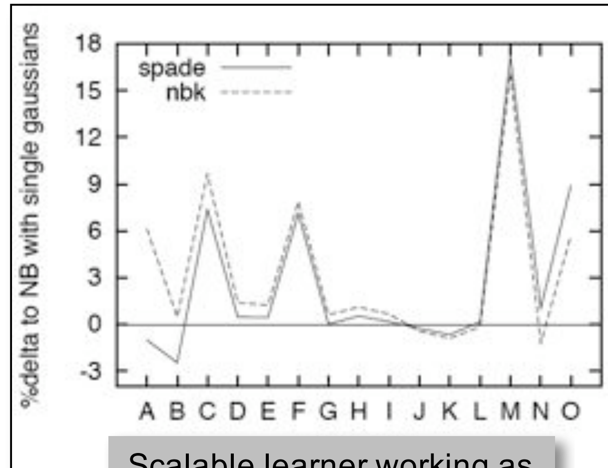
- **Data comes from NASA**
  - Process metrics:
    - JPL project data
    - IV&V effort potential data
  - Product metrics
    - Defect logs from multiple NASA centers
    - Flight simulator data
- **Conclusions apply to NASA projects**



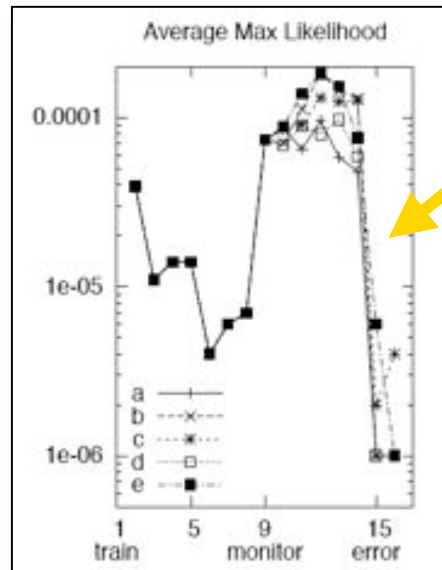
project	# modules	% with defects	language	developed at	notes
CM1	496	9.7%	C	location 2	a NASA spacecraft instrument
JM1	10885	19%	C	location 3	real-time predictive ground system: uses simulations to generate the predictions
KC1	2107	15.4%	C++	location 4	storage management for receiving and processing ground data
KC2	523	20%	C++	location 4	science data processing; another part of the same project as KC1; different personnel to KC1. shared some third-party software libraries as KC1, but no other software overlap.
PC1	1107	6.8	C	location 5	support tools
Total	15118				

Accomplishments

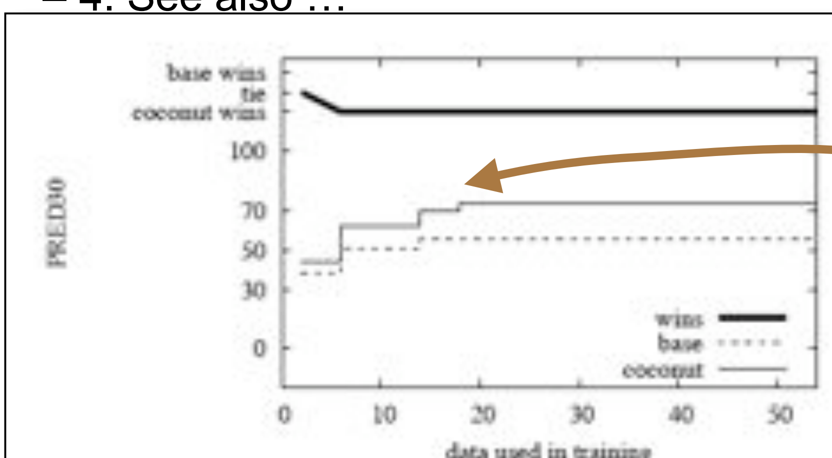
- **Before:**
  - –Can automatically learn defect detectors from error logs
  - –Those defect detectors from code are much BETTER than previously believed
    - Yes, false negative, but adequate to good detection probabilities
    - (Enough) stability across multiple projects
- **Now:**
  - –1. Can automatically learn software cost models
    - AND determine how much data is required to do that
  - –2. Can scale up to HUGE data sets
  - –3. Can determine when a learned theory goes "out of scope"
  - – 4. See also …



Scalable learner working as well as state-of-the-art, non-scalable, alternative



Where a learner has left the zone where it was certified



When we have seen enough data to learn a good cost model

1. Can automatically learn software cost models AND determine how much data is required to do that

Research Heaven, West Virginia

- **Late 2004:**
  - Much work on learning software cost models
- **Early 2005:**
  - That work transferred to a separate SARP project
  - “How much will it cost”?
- **Before the transfer (see next slide…)**

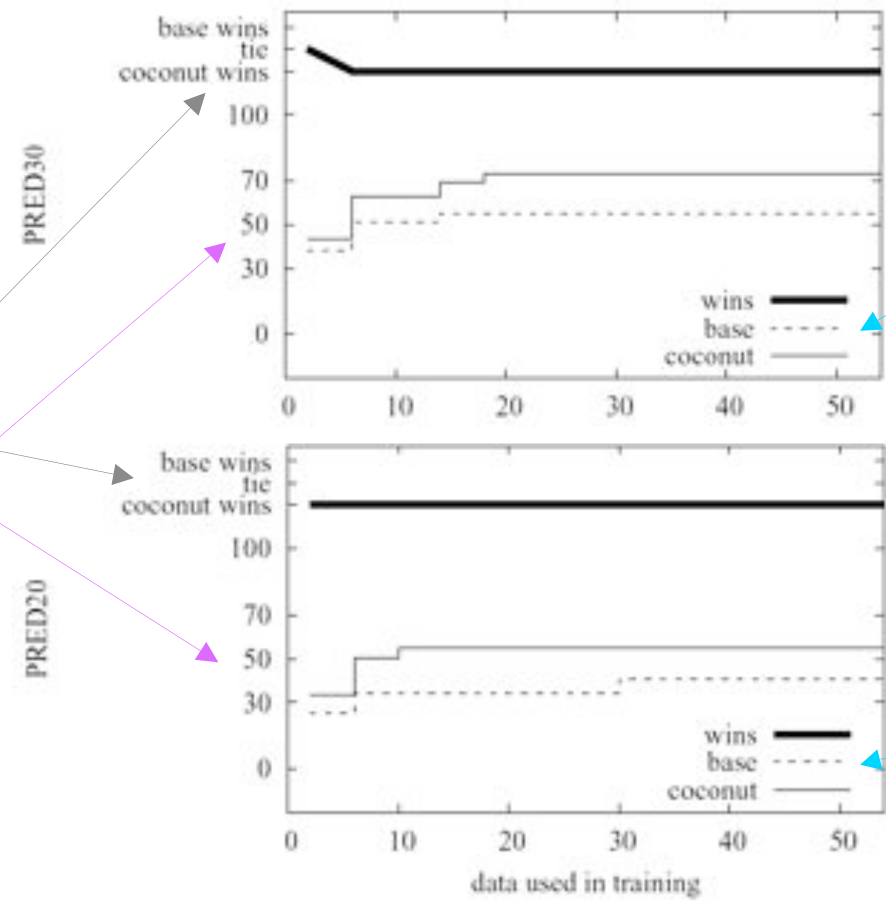
Straw man

60 NASA COCOMO-I Projects in PROMISE repository

base = $a*sloc^b$

cocomo81 = $a*sloc^b * em_1 * em_2 *\ldots$

- **30 repeats (randomizing the order)**
- **Use t-tests to compare**
  - PRED(N) using coc81 or base
  - PRED(N) after N1 or N2 projects
- **Significant changes up to**
  - 18 projects for PRED(30)
  - 30 projects for PRED(20)



2. Can scale up to HUGE data sets

- **Work with Andres Orrego (TMC)**
- **Bayes classifiers**

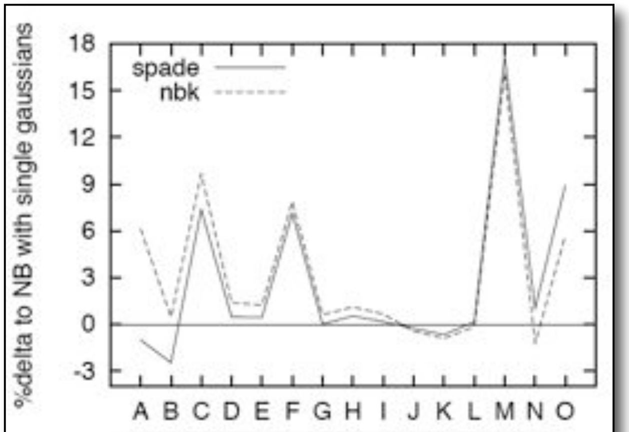
	E_1	E_2	E_3
$H = car$	job	suburb	wealthy?
ford	tailor	NW	y
ford	tailor	SE	n
ford	tinker	SE	n
bmw	tinker	NW	y
bmw	tinker	NW	y
bmw	tailor	NW	y

	$P(E_i \mid H)$		
P(H)	job	suburb	wealthy?
ford:3=0.5	tinker:1=0.33	NW:1=0.33	y:1=0.33
	tailor:2=0.67	SE:2=0.67	n:2=0.67
bmw:3=0.5	tinker:2=0.67	NW:3=1.00	y:3=1.00
	tailor:1=0.33	SE:0=0.00	n:0=0.00

- L(H | E) = P(E | H) * P(H)
- P(H | E) = L(H | E) / sumOfAllLiklihoods
- E.g. L(bmw| job=tinker and suburb=NW)= $\boxed{0.33}$ * $\boxed{1.00}$ * $\boxed{0.5}$ = 0.165
- Incremental, fast learning, fast classification, small memory footprint
- Some issues with dependencies but, in practice, works well
- But assume non-numeric data

Numerics and Bayes

- **Kernel functions**
  - Gaussian (standard)
  - Kernel estimation (John & Langley)
  - etc
- **Discretization policies**
  - N-bins: (max-min)/N
  - Bin Logging,
  - Etc
- **All N-pass methods**
  - And scalable data miners should be one pass
- **SPADE:**
  - Incremental N-bins
  - Simple++, one-pass
  - Works very well.



Scalable learner working as well as state-of-the-art, non-scalable, alternative

When enough is enough

Research Heaven, West Virginia

For 20 data sets and learners, plateau after a few 100 examples

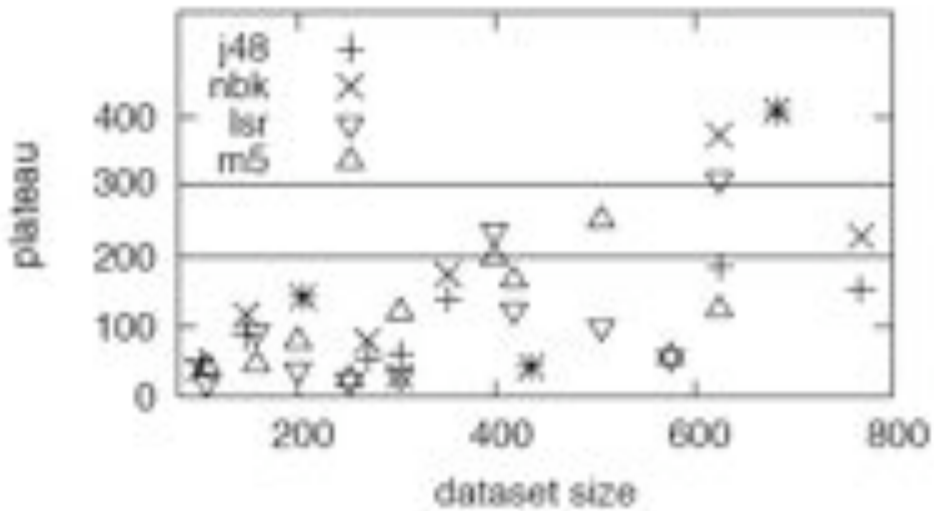
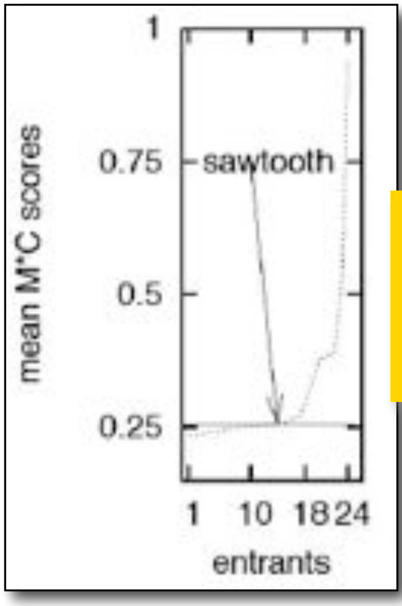


Fig. 1. *10*10* incremental cross validation experiments with J48 and Naive-Bayes (with kernel estimation) on {A:heart-c, B:zoo; C:vote; D:heart-statlog; E:lymph, F:autos, G:ionosphere, H:diabetes, I:balance-scale, J:soybean}; M5 and LSR on {K:bodyfat, L:cloud, M:fishcatch, N:sensory, O:pwLinear, Q:strike, R:pbc, S:autoMpg, T:housing}. All data sets from the UCI repository [8]. Data sets A..J have discrete classes and are scored via the *accuracy* of the learned theory; i.e % successful classifications. Data sets K..T have continuous classes and are scored by the *PRED(30)* of the learned theory; i.e. what % of the estimated values are within 30% of the actual value.

SAWTOOTH= plateau + SPADE

Research Heaven, West Virginia

- **Learn till plateau**
- **Only start learning again if performance falls off plateau**
  - Recognition of mode changes
- **KDD data (5,000,000 examples):**
- **In summary:**
  - Now we can see a lot, learn a little,tell just enough

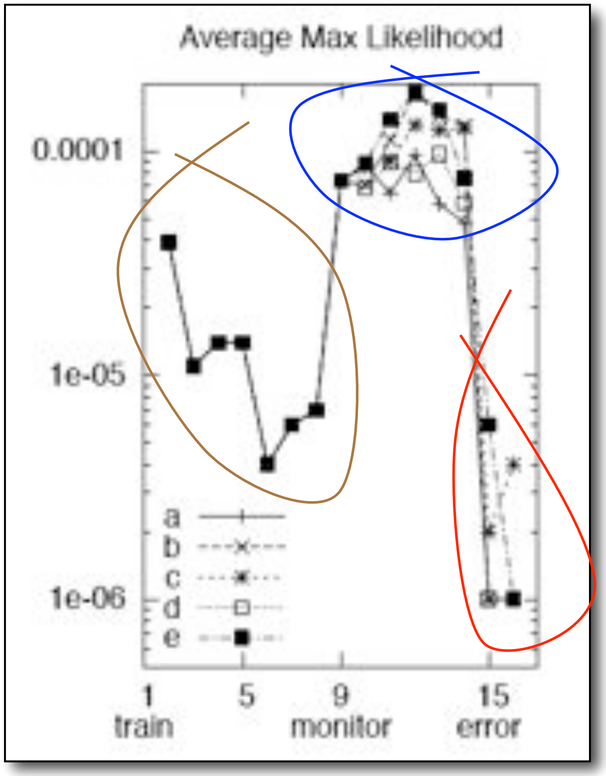


Compares well to sate of the art methods

3. Can determine when a learned theory goes "out of scope"

Research Heaven, West Virginia

- **AI & learning & validation**
  - Monday:
    - System is certified
  - Tuesday:
    - Launch
  - Wednesday:
    - AI learner adapts the software
    - Does the old certification hold?
- **Solution:**
  - Anomaly detection
  - Detect when new context out of scope of prior certification
  - Rings the alarm bells, tells pilot to eject, calls the tiger teams, places device into sleep mode
  - Many previous (complex) solutions
  - Very simple in a SAWTOOTH/SPADE context
    - Place all examples in one class
    - Track average likelihood of new examples in that class



Commissioning
Normal operation
Abnormal situation

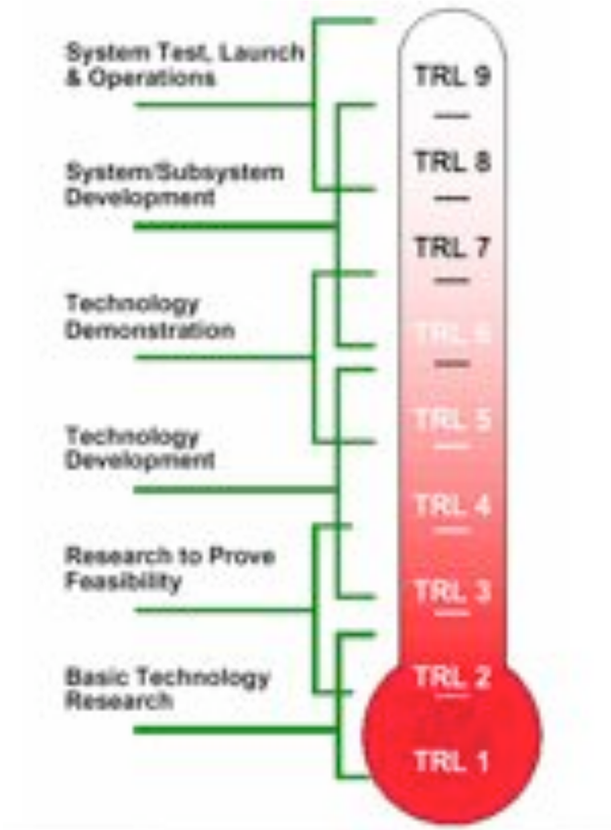
4. See Also…

Research Heaven,
West Virginia

- **Much related SARP work**
- **“Martha”:**
  - Spot/Cube
- **“Tandem Experiments”:**
  - SPY
- **“How much will it cost”:**
  - Learning software cost models
- **“GSFC metrics project”**
  - Giving tools to users

Research Heaven,
West Virginia

Technology Readiness Level of the Work = 5 or 6



- **5:**
  - Component and/or breadboard validation in a relevant environment
- **6:**
  - System/subsystem model or prototype demonstration in a relevant environment (Ground or Space)

Potential Applications

Research Heaven,
West Virginia

- **1. Can automatically learn software cost models AND determine how much data is required to do that**
  - Software cost estimation
  - Generating locally relevant estimates
- **2. Can scale up to HUGE data sets**
  - Simulation-based acquisition
  - Any simulator-based analysis
- **3. Can determine when a learned theory goes "out of scope"**
  - Certification and runtime monitoring of autonomous systems

Availability of data or case studies

Research Heaven, West Virginia

- **Data**
  - Cost estimation data sets public:
    - http://promise.site.uottawa.ca/SERepository/datasets/cocomo81.arff
    - http://promise.site.uottawa.ca/SERepository/datasets/cocomonasa_v1.arff
  - Other datasets proprietary
- **Software:**
  - Free, on request

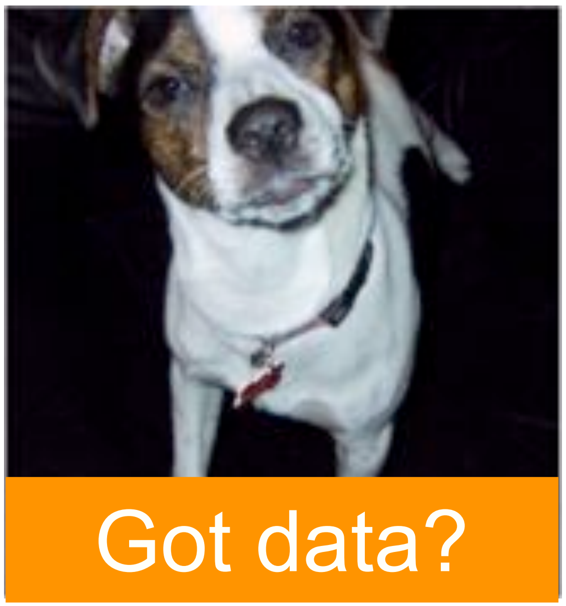
Barriers to research or applications

Research Heaven, West Virginia

- **Getting data**
- **Nervousness regarding use of AI learning systems**
  - Good news: much recent NASA work on ISHMs

Next Steps

Research Heaven,
West Virginia



- **Data mining needs data**
  - Got data?
    - Then meet your new best friend
- **Current plans**
  - More defect data studies
    - Dozens, not just 5, data sets
    - Check effectiveness and stability?
  - Release of the generalized toolkits
    - Tutorials
    - manuals
  - Generalized anomaly detectors
    - The “selection bias” problem
  - Synergies with other SARP data mining projects